



Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

Bescheinigung

Certificate

Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

jc841 U.S. PTO

09/759022



01/11/01

Patentanmeldung Nr. Patent application No. Demande de brevet n°

00200161.8

Der Präsident des Europäischen Patentamts:
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

I.L.C. HATTEN-HECKMAN

DEN HAAG, DEN
THE HAGUE,
LA HAYE, LE

17/04/00

This Page Blank (uspto)



Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

PH 117 600 001
LIS

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:
Demande n°: 00200161.8

Anmeldetag:
Date of filing:
Date de dépôt: 17/01/00

Anmelder:
Applicant(s):
Demandeur(s):
Koninklijke Philips Electronics N.V.
5621 BA Eindhoven
NETHERLANDS

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
Text improvement

In Anspruch genommene Priorität(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat:
State:
Pays:

Tag:
Date:
Date:

Aktenzeichen:
File no.
Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:

/

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE
Etats contractants désignés lors du dépôt:

Bemerkungen:
Remarks:
Remarques:

This page Blank (uspto)

Text improvement.

The invention relates to a method and device for text improvement.

Nowadays digital display devices are more and more frequently matrix devices, e.g. Liquid Crystal Displays, where each pixel is mapped on a location of the screen having a one to one relationship between raster data and display's points. This technology implies the usage of a scaling system to change the format of the input video/graphic signal so that it satisfies the size of the device, i.e. the number of its pixels. The scaling block is based on a filter bank that performs pixel interpolation when the zooming factor is varying. Actually available solutions on the market apply an undifferentiated processing on the graphic raster that leads to results with unavoidable artifacts. Usually low-pass filters reduce pixellation, also know as the seesaw effect on diagonals, and prevent the signal to suffer from aliasing due to the sub-sampling, but they also introduce other annoying effects such as blurring the images. It depends on the content of the displayed signal the relevance of the perceived artifacts and the kind of artifacts that have to be preferred as unavoidable.

It is, inter alia, an object of the invention to provide a simple text improvement. To this end the invention provides a text improvement as defined in the independent claims. Advantages embodiments are defined in the dependent claims.

Starting from the above-mentioned observations, a novel technique is provided here that is able to take into account the image content and to apply an ad hoc post-processing only where it is required. So, in accordance with the present invention, text improvement is based on detection. The processing is active only in presence of text region. A viable area of application of this invention is the text readability improvement in the case of LCD devices, when, and it is usually the case, we do not want to affect other parts of the displayed signal.

A remarkable characteristic of the technique presented here is its really low computational complexity. This aspect determines a high effectiveness in terms of cost/performances ratio. In fact the insertion of the proposed algorithm into the other circuitry that carries out all the digital processing needed for resize the matrix display device input,

presumably rises the display quality, according with the average user perception, without affecting considerable its cost.

These and other aspect of the invention will be apparent from and elucidated with reference to the embodiments described hereinafter.

5

In the drawings:

Figs. 1-3 illustrate the operation of a morphological filter; and

Fig. 4 shows a block diagram of a system in accordance with the present invention.

10

The invention proposes the design of a text detection algorithm, together with a post-processing block, for text enhancement. It will be shown that the invention significantly improves the performance in terms of content readability and leads to good perceptual results of the whole displayed signal, while keeping really low the computational complexity of the total scaling system.

15

The organization of the remainder of this document is as follows. First the general scaling problem and the current available algorithms are briefly summarized. Thereafter concepts concerning format conversion by a non-integer factor will be introduced. Successively the post-processing block, characterized by the thresholding operation and the morphological filter, will be summarized and its features will be described. Finally the text search strategy will be presented and the detection algorithm and its cooperation with the previously introduced post-processing block will be elucidated.

20

The general framework

25

Resizing pictures into a different scale requires format conversion. This operation involves well-known re-sampling theory and classical filter procedures are currently used to accomplish with it. Filters avoid aliasing problems in frequency, freeing room for the repetitions introduced by the sampling operation in the original domain. Among the interpolation filter families polynomial interpolators of first order are commonly used in which the reconstructed pixel is a weighted mean of the nearest pixels values. These kinds of filters are also called Finite Impulse Response filters.

30

Inside standard display devices the format conversion problem is usually faced with linear filtering too. A particularly simple class of F.I.R. filters reconstructs pixels in between two available ones tacking the value on the line joining these two adjacent points.

There are many other possible techniques. For example pixel repetition, or polynomial interpolation with more complexes weighting functions. The quality perception of images processed with these different solutions is generally not really high, there are impairments and artifacts that are not completely avoidable. This consideration implies that some compromise is due in order to reach an acceptable or, at the best, a satisfactory cost/performance ratio.

In the past the simplest solution solved the problem using pixel repetition. A more recent solution, see the Philips scaler PS6721, still uses linear filtering but with a slight different shape of the impulse response, to improve the transition steepness.

Measuring the rise time of the step response is a classical way to assess the performance of the interpolator in presence of an edge. In fact low pass filters affect edge steepness and a smooth steepness is perceived as a blurring effect.

Moreover the actual impact of this annoying artifact depends on the kind of displayed signal. Actually in case of natural images a blurring effect could be tolerated in a certain measure.

Whereas for artificial pattern a slight smoothness effect is recommended only when the content requires approaching a natural impression (this is the case of virtual reality and 3D games). In this case filtering is used as an anti-aliasing process. For the same reason these kinds of filters are used on text/characters to avoid the pixellation effect, also know as the seesaw effect on diagonals. Interpolation filters are anti-aliasing filters too, because they reduce the highest frequency of the input signal. Moreover, supposing to have a black text on a white background, the amount of gray levels introduced by this kind of filters should be a less percentage of the black quote. If it is not the case, we have an artifact instead of a picture improvement, and images are perceived as blurred. For instance, when bilinear interpolation as well more sophisticated filters like the bicubic ones, are used on small characters (the commonly used size 10÷12 points) and thin lines, they appear defocused. In all these cases it seems better to use no filters at all, at least no low pass filters as are actually available.

Starting from the above consideration we can conclude that, because format conversion requires resampling, so that the filtering process is unavoidable, to accomplish with the above issue we have to find out some other solution. In case of text, a simple idea is to apply a post-processing block after the scaler to clean all the gray levels where characters are detected. Because of the scale change, this operation could not be performed using only a simple threshold block. In fact threshold is a non-linear operator that introduces not uniform patterns when it converts gray levels characters to binary values, another kind of artifact that

is highly noticeable. Morphological filters are an interesting class of operators that are able to change not regular patterns into more regular ones. They will be introduced in a following section.

5 Format conversion by a rational factor

In today's digital display devices, images are frequently represented with a matrix of pixels so that a fixed picture format is required. When a signal with a different format arrives at the input of a matrix display, format conversion is unavoidable. Supposing a graphic card had generated the signal, than selecting a different graphic format, instead of the one used by the display, depends on the requirement of the software application running. At the moment it is not advisable to constraint the graphic card output only with the requirement of the display.

We recall that standard today's graphic formats are VGA, SVGA, XGA, SXGA and higher. Format conversion between these raster sizes requires in almost all the cases a rescaling by a rational factor. This, by itself, leads to a sensible degradation of the resampled picture. In fact when, for example, we need to change format from VGA at display's input to XGA at display's output, the factor involved would be 8/5, equal to 1.6 times the size of the original picture. This format conversion ratio would clearly require a sub-pixel resolution, but with standard linear filtering techniques this is not be allowed without paying a high blurring cost.

Let $s(i, j)$ be the input signal at position (i, j) in the input grid and $\hat{s}(\tilde{i}, \tilde{j})$ the signal after format conversion at position (\tilde{i}, \tilde{j}) in the thicker output grid. Supposing to have a rescaling from VGA to XGA, i.e. by an 8/5 factor, every 5 pixels at the input of the sampler there will be 8 pixels at its output. A rescaling by a rational factor conceptually relies on an intermediate "super-resolution" grid obtained using a zooming factor equal to the numerator, in the example 8. In this case the "super-resolution" grid will be eight times thicker than the original one. Tacking two input values, $s(i, j)$ and $s(i+1, j)$, on the same line j at position i and $i+1$, the interpolated value $\hat{s}(\tilde{i}, \tilde{j})$ will be positioned in between the two original values in the super-resolution grid, i.e. in one of the eight possible positions available on the grid.

We express this fact by the following equation:

$$\hat{s}(i + \frac{k}{8}, j) = w_1 \cdot s(i, j) + w_2 \cdot s(i+1, j) \quad \forall k \in [0 \dots 7]$$

Where, for the linear interpolator,

$$\begin{cases} w_1 = \delta \\ w_2 = 1 - \delta \end{cases}$$

- 5 k is the position of the pixel in the dense grid, the position is also called filter phase. In a linear filter is $\delta \propto k$, and δ is the distance between the pixel to be interpolated respect to the two adjacent original ones. The signal at the output grid will be obtained tacking values on a grid 5 times weaker. The sub-sampled signal at the output is expressed as it follows:

10
$$s^*(i + 5 \cdot \frac{k}{8}, j) = w_1 \cdot \hat{s}(i + \frac{\bar{k}}{8}, j) \quad \bar{k} \in [0 \dots 7]$$

- Because the output grid is not a multiple of the input grid, often original pixel value will be lost and they will be replaced with an average value according with what we said above. If the input pattern would be a black and white text, its pixels will be frequently replaced by a weighted average of their values, a gray level.
- 15

Text improvement via thresholding

- A threshold operator placed at the output of the scaling filter will recover a black and white pattern, or more generally a bicolor one, choosing the threshold nearest value according to the following relationship:
- 20

$$\begin{cases} s_g^*(i + \frac{8}{5} \cdot k, j) = l_k & \text{if } s^*(i + \frac{8}{5} \cdot k, j) < \vartheta \\ s_g^*(i + \frac{8}{5} \cdot k, j) = l_w & \text{if } s^*(i + \frac{8}{5} \cdot k, j) \geq \vartheta \end{cases}$$

Where l_k is the black level and l_w the white one.

- 25 We could notice that, in case of black/white and bicolor patterns, the threshold function could be integrated in the filter operator, setting l_k and l_w in accordance with the actual filter phase. In this way the threshold operation recover original bicolor levels from the interpolated ones according with theirs new positions. In regions where the amount of gray levels introduced is too height, this simple operator improves the sharp edge perception.

Anyway this is paid with the introduction of irregular patterns. In the next section we will see how this problem could be solved.

Morphological filtering algorithms

- 5 The introduction of mathematical morphology to solve the problem of text deblurring is due to the fact that a morphological filter, working both as a detector and as a non-linear operator, is able to eliminate gray levels without destroying the character regularity. Moreover, in case of bicolor patterns, is able to recover a specified regularity where required.
- 10 In general the detector, called structuring element, is a small matrix (usually 2x2 or 3x3); it can recognize a particular pattern on the data, in our case the rasterized image's pixels at the display output, and to substitute that pattern with a different set of requested values. Supposing to use the morphological filter after the threshold block, on a bi-level pattern, the structuring element will work as a binary mask on the underlying data performing a set of
- 15 logical operations between the bit of the running matrix and the bit of the scanned data. An output equal at 1 will signify that a specified pattern has been identified.

A particular operator belonging to the morphological filter family, also called "diagonal" filter, applies the following set of logical operations to the data:

$$\begin{aligned}
 Y &= X_4 \cup (P_1 \cup P_2 \cup P_3 \cup P_4) \\
 P_1 &= (X_4^c \cap X_7 \cap X_6^c \cap X_3) \\
 P_2 &= (X_4^c \cap X_3 \cap X_0^c \cap X_1) \\
 P_3 &= (X_4^c \cap X_1 \cap X_2^c \cap X_5) \\
 P_4 &= (X_4^c \cap X_5 \cap X_8^c \cap X_7)
 \end{aligned}$$

20

Here, $X_0 \dots X_8$ is the set of data currently analyzed by the structuring element; besides, in case of binary data, \cup is the classic logical **OR** operator and \cap is the classic logical **AND** operator. The structuring element orders the data in its framework as shown in Fig. 1.

25

The output, y , after the set of logical operations above introduced, replaces the previous value at the origin of the data matrix, X_4 in the figure. You can notice that, if the result of

$P_1 \cup P_2 \cup P_3 \cup P_4$ is 0, than X_4 remains unchanged, instead if the result is 1 than X_4 is always replaced by 1.

Looking carefully, it will be evident that the set P_1, P_2, P_3, P_4 of logical operations corresponds to the detection of the patterns shown in Fig. 2. Patterns in Fig. 2 are diagonal patterns of black and white pixels, in case of binary images. According with the above relations, when one of these configurations is found, than, in the origin of the detected region identified by a circle in the figure, a 0 value is substitute by a 1. This operation, in terms of pattern effect, fills holes in diagonal configurations.

One could notice that the same operation could be done, instead of using logical operators, with a LUT addressed by the configuration of bits in the structuring element. Supposing to order the cells of the element according with the above figure, this configuration has the following address:

$$LUT_{address} = X_8 X_7 X_6 X_5 X_4 X_3 X_2 X_1 X_0$$

where each X_i is correspondingly equal at 1 or 0 according with the value in the i^{th} position of the matrix. To fill holes, the LUT at position $XXX10X01X$, $01X10XXXX$, $X10X01XXX$, $XXXX01X10$, will be set at 1, in all the other position it will be set at 0. Here X means don't care.

From a conceptual point of view, because of the holes filling function of the "diagonal" structuring element, the set of operations described above on a 3×3 structuring element, are equivalent at changing a diagonal patterns, anyhow oriented in a 2×2 matrix, with a uniform block. This concept is clarified in Fig. 3.

Block diagram of system embodiment

The drawing in Fig. 4 shows a block diagram of the total system in which the main concept of the architecture for the detector and the post processing block are sketched. An input image $InIm s$ is applied to a Search Window part SW and Text Detector part Det. The input image $InIm s$, possibly modified in some region by the text detector part Det, is applied to a scaler Scal, if recognized as text by the text detector part Det, such as the commercially available scaler PS6721. The scaled image from the scaler Scal is applied to a post-processing part Post-proc that produces the output image $OutIm s^*$.

Search Window and Text Detector

Detector is a key operator. In fact it depends on it if the input signal will be binarized and further processed or simply filtered with the linear scaler. According to what it was previously said, detection is specifically designed to recognize text patterns. When the
5 required constraint imposed at the detector are not satisfied, the signal does not eventually benefit of this further processing step. Detection is performed with a local sensor that recognizes the amount of colors in a small region. So in principle it works as a search window that scan the raster image to discover text areas.

To design it, satisfying a low memory cost, a fixed vertical width was used,
10 equal to 3 lines on the domain of the original signal. Instead its horizontal depth is varying according to the image characteristics and it is based on a simple growth criterion defined using some intuitive assumptions on the text properties. Currently assumptions on graphic text are as it follows:

- 15 1. *A text area is a two-color region in which text is one color and the other color is the background.*
2. *In a text area a text color is perceptually fewer present than a background color.*
3. *A text region has a reasonable horizontal extension.*

20 These assumptions determine the constraints on the patterns the detector recognize as text regions. As we can see neither filtered text nor not-uniform background are recognized as text regions. This is a reasonable assumption because the threshold operator in these cases would introduce more artifacts than benefits. Furthermore the not balanced
25 percentage of the two colors prevents the detector from identifying as text two color regions with potentially dangerous patterns. An example is the chess pattern, quite recurrent, for example in window folder background. Finally the third condition prevents to identify as text regions small bicolor fragment of the raster signal, that could be presumably border or other small pieces of graphic objects.

The conditions introduced above are used to define some parameters to adjust
30 the behavior of the detector such that it could reach the best performances. Let we consider the above introduced input raster signal $s(r, c)$ at position (r, c) . The search window will be indicated with $q(r, c)$, with (r, c) being the coordinates of the block's origin that identify its reference pixel in the image; the relative coordinates, identifying a cell in the search window, are referred to the block origin and they will be noted as (i, j) . Furthermore the detector

height and width will be indicated with h and w . Whereas w is a varying parameter, on the contrary h is fixed, to satisfy line memory constraints, and its value is currently $h = \bar{h} = 3$.

Let be N_c the number of colors detected in the search window. According with the previously described block growing, the width w will increase following this search strategy:

$$\begin{cases} w(k+1) = w(k) + 1 & \text{if } N_c \leq 2 \\ w(k+1) = w(k) = w & \text{if } N_c > 2 \end{cases}$$

$N_c > 2$ is the exit condition from the growing search strategy. When the exit condition is verified, the system will return the final block width w .

Together with the block growing process, two color counters will be incremented at each new step k . It could be notice that a step k corresponds to the evaluation of a new input pixel in the horizontal direction. Calling γ_1 the number of pixels with color c_1 and γ_2 the number of pixels with color c_2 , the counters will be upgraded according with the corresponding block growing step in the following manner:

$$\begin{cases} \gamma_1(\tau+1) = \gamma_1(\tau) + 1 & \text{if } q(i + w(k+1), j + h) = c_1 \text{ for } h = 1 \dots 3 \\ \gamma_1(\tau+1) = \gamma_1(\tau) & \text{otherwise} \end{cases}$$

and

$$\begin{cases} \gamma_2(\tau+1) = \gamma_2(\tau) + 1 & \text{if } q(i + w(k+1), j + h) = c_2 \text{ for } h = 1 \dots 3 \\ \gamma_2(\tau+1) = \gamma_2(\tau) & \text{otherwise} \end{cases}$$

$\tau = 3 \cdot w(k+1) + h$ is a new counting step in the search window, a new pixel evaluated using the growing window at item k .

Finally let we introduce the last parameter ξ , representing the ratio between the two colors counters, once the background is identified, according to the following relationship:

$$\begin{cases} \xi = \frac{\gamma_1}{\gamma_2} & \text{if } \gamma_1 \geq \gamma_2 \Rightarrow c_1 = \text{background} \\ \xi = \frac{\gamma_2}{\gamma_1} & \text{if } \gamma_1 < \gamma_2 \Rightarrow c_2 = \text{background} \end{cases}$$

Once the algorithm is exited from the search strategy, the detection window is available to identify its content.

- 5 As mentioned above, a first condition to be satisfied, so that a region would be recognized as text, is that the block has a reasonable extension. Let be:

$$\bar{\varepsilon} = \min w$$

- 10 the minimum value, in terms of pixels, allowed for a region to be recognized as text region. The condition to be satisfied by a text region will be:

$$w \geq \bar{\varepsilon}$$

- 15 The current value fixed for the parameter $\bar{\varepsilon}$ is: $\bar{\varepsilon} = 300$.

Recalling that ξ is the ratio between the background and the text colors, a second condition to be satisfied, so that the block would be recognized as a text area will be:

$$\xi \geq \bar{\xi}$$

20

where $\bar{\xi}$ is a modifiable parameter actually fixed as $\bar{\xi} = 1.2$. In other words:

$$\text{if } \xi < \bar{\xi} \Rightarrow q[\cdot] \text{ not a text window}$$

- 25 The block will be discarded as not a text block when one of the above conditions are not satisfied. The new search window will be $q(r, c + w)$ and it will start at position $(r, c + w)$ in the original image, or $(r + 3, c)$ depending if in the previous step the end of line was reached.

- Following this strategy the entire image will be scanned by the search window and text region will be detected. As text is detected the previously described post-processing operations will be applied.
- 30

Going back to Fig. 4, an input image is first subjected to a block-growing process BlGr based on whether the number of different colors does not exceed 2 ($N_c \leq 2$), a first indication for the presence of text. As soon as the number of colors exceeds 2, the block growing process BlGr is stopped, and the other parameters Outpar are determined, which represent the three criteria for text listed above. Based on these parameters Outpar, it is determined whether there is a text region (Txt reg ?). If so, the background color $c_{\text{background}}$ is set to white, and the text color c_{text} is set to black.

The resulting image is subjected to the scaling operation SCAL.

After the scaling operation SCAL, the text region is subjected to a thresholding operation (threshold 9), the output of which is applied to a morphological filter (Morph. Filt.). Thereafter, white is set back to the background color $c_{\text{background}}$, and black is set back to the text color c_{text} . The result of this operation forms the output image OutIm s* that is displayed on a matrix display D.

A primary aspect of the invention can be summarized as follows. A novel technique is suggested able to take into account the image content and to apply an ad-hoc scaler post-processing only where it is requested. A viable area of application of this invention is the text readability improvement in the case of LCD devices, when, and it is usually the case, we do not want to affect other part of the displayed signal. It is, inter alia, an object of the invention to provide an ad hoc simple text detector. The invention proposes the design of a text detection algorithm, together with a post-processing block, for text enhancement. The invention significantly improves the performance in terms of content readability and leads to good perceptual results of the whole displayed signal, while keeping really low the computational complexity of the total scaling system. The invention is preferably applied in LCD scaler ICs.

It should be noted that the above-mentioned embodiments illustrate rather than limit the invention, and that those skilled in the art will be able to design many alternative embodiments without departing from the scope of the appended claims. In the claims, any reference signs placed between parentheses shall not be construed as limiting the claim. The word "comprising" does not exclude the presence of elements or steps other than those listed in a claim. The word "a" or "an" preceding an element does not exclude the presence of a plurality of such elements. The invention can be implemented by means of hardware

comprising several distinct elements, and by means of a suitably programmed computer. In the device claim enumerating several means, several of these means can be embodied by one and the same item of hardware. The mere fact that certain measures are recited in mutually different dependent claims does not indicate that a combination of these measures cannot be used to advantage.

References:

1. W.K. Pratt, "*CAP 15: Morphological Image Processing*" in "*Digital Image Processing*", J. Wiley & Sons, 1991.
- 10 2. A.V. Oppenheim, R.W. Schaffer, *Discrete-Time Signal Processing*, Prentice-Hall International, Inc. 1989.
3. P. Zamperoni, *Metodi dell'elaborazione digitale di immagini*, Masson, Milano, 1990
4. H.J.A.M. Heijmans, C. Ronse, *The Algebraic Basis of Mathematical Morphology I. Dilations and Erosions*, Comput. Vision Graphics Image Process. 50, pp. 245-295, 1990.
- 15 5. H.J.A.M. Heijmans, C. Ronse, *The Algebraic Basis of Mathematical Morphology II. Openings and Closings*, CVGIP: Image Understanding, Vol. 54, No.1, pp. 74-97, 1991.
6. E.R. Dougherty, *Morphological Image Processing*, SPIE Optical Engineering Press, 1992
7. S.R. Sternberg, R.M. Haralick, X. Zhuang, *Image Analysis Using Mathematical Morphology*, IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-9, No. 4, pp. 532-
20 550, July 1987

CLAIMS:

1. A method of text improvement, the method comprising the steps of:
detecting (SW, Det) text in an image; and
processing (Post-proc) the image in dependence on a result of the text
detecting step.
5
2. A method as claimed in claim 1, wherein the detecting step (SW, Det)
comprises the step of determining (BlGr) a region for which it holds that the number of
colors does not exceed 2.
- 10 3. A method as claimed in claim 1, wherein the detecting step (SW, Det)
comprises the step of determining whether a text color is fewer present than a background
color.
4. A method as claimed in claim 1, wherein between the detecting step (SW, Det)
15 and the processing step (Post-proc), a scaling step (Scal) is carried out to adjust first numbers
of pixels per line and lines per image of the image to second numbers of pixels per line and
lines per image that fit in with a display (D) on which the image is displayed.
5. A method as claimed in claim 4, wherein the processing step (Post-proc)
20 comprises the step of subjecting a scaled image to a thresholding operation.
6. A method as claimed in claim 4, wherein the processing step (Post-proc)
comprises the step of subjecting a scaled image to a morphological filtering.
- 25 7. A method as claimed in claim 4, wherein the detecting step (SW, Det)
comprises the step of setting a background color ($c_{\text{background}}$) to white, and a text color (c_{text}) to
black, and the processing step (Post-proc) comprises the step of setting white back to the
background color ($c_{\text{background}}$), and black back to the text color (c_{text}).

8. A device for text improvement, the device comprising:
means for detecting (SW, Det) text in an image; and
means for processing (Post-proc) the image in dependence on a result of the
text detecting means.

5

9. A display apparatus, comprising:
a device for text improvement as claimed in claim 8; and
a display (D).

ABSTRACT:

In a method of text improvement, in an image text is detected (SW, Det), and the image is processed (Post-proc) in dependence on a result of the text detection.

(Fig. 4)

This Page Blank (uspto)

1/2

x_0	x_3	x_6
x_1	x_4	x_7
x_2	x_5	x_8

FIG. 1

?	1	0	0	1	?	?	?	?	?	?	?
?	0	1	1	0	?	1	0	?	?	0	1
?	?	?	?	?	?	0	1	?	?	1	0

FIG. 2

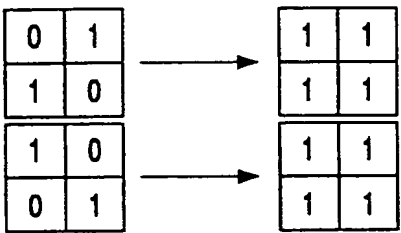


FIG. 3

2/2

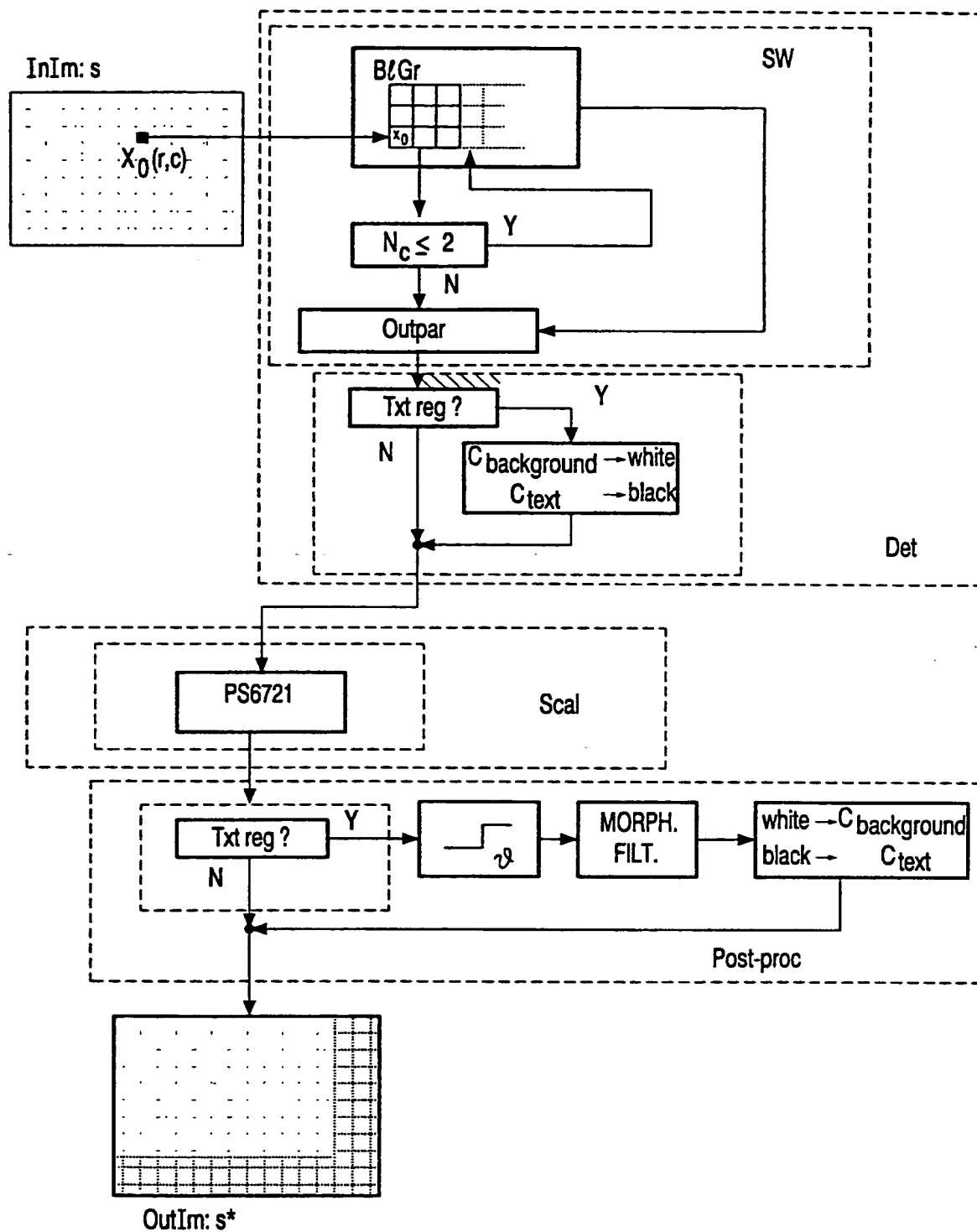


FIG. 4